A STUDY OF COMPUTER BASED DATA SECURITY TECHNIQUES

William Lloydd Lupton

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

A Study of
Computer Based
Data Security Techniques

by

William Lloydd Lupton

Thesis Advisor: G. L. Barksdale, Jr.

Approved for public release; distribution unlimited.

A Study Of Computer Based Data Security Techniques

by

William Lloydd Lupton
Lieutenant, United States Navy
B. S. Naval Postgraduate School, 1972

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN COMPUTER SCIENCE

from the

NAVAL POSTGRADUATE SCHOOL
June, 1973

ABSTRACT

This essay presents the results of a study which surveyed the various aspects of system security hardware, software, and procedural techniques in use in current and proposed automated systems. Its impetus is from the concern for security control that has been generated by the increasing number of time-sharing and resource-sharing systems. The intention is to present the designers, managers, programmers, system implementers, and operational personnel with a consolidated source of data concerning security techniques and with a tool to evaluate the data and select the techniques applicable to their respective security requirements.

TABLE OF CONTENTS

I. INTRODUCTION 6
II. NATURE OF THE PROBLEM 7
III. SECURITY REQUIREMENTS-GENERAL 8
A. MAINTENANCE AND SUPPORT ACCESS..... 8
B. FAILURE ACCESS..... 10
C. DELIBERATE PASSIVE ACCESS..... 11
IV. SECURITY REQUIREMENTS-PARTICULAR 13
A. USER LANGUAGE CAPABILITY..... 13
B. TERMINAL LOCATION AND USAGE..... 14
C. DATA CLASSIFICATION..... 14
V. COMPARISON BETWEEN TECHNIQUES 17
A. SOFTWARE TECHNIQUES..... 18
1. User Interface..... 18
2. Terminal Subsystem..... 26
3. Executive/Monitor..... 28
4. File Handler..... 32
5. Others..... 37
B. HARDWARE TECHNIQUES..... 39
1. Central processor Unit..... 39
2. Main Memory Module..... 46
3. I/O Control processor..... 48
4. Direct Access Memory (DAM)..... 50
5. Remote and Local User terminals..... 51
6. General Techniques..... 52
C. HARDWARE AND SOFTWARE COMPARISONS..... 53
D. PROCEDURAL TECHNIQUES..... 54
1. Classifying and Declassifying Procedures..... 58
2. Safeguarding Procedures..... 58
3. Accountability..... 59
4. Dissemination..... 59
E. MANUAL AND AUTOMATED PROCEDURE COMPARISON..... 59
VI. CONCLUSIONS 62

LIST OF FIGURES

1. THREAT POINTS..... 9
2. FEATURES THAT DETERMINE SYSTEM SECURITY..... 12
3. USER CAPABILITY..... 15
4. SOFTWARE TECHNIQUE SUMMARY..... 19
5. HARDWARE TECHNIQUE SUMMARY..... 40
6. HARDWARE AND SOFTWARE TECHNIQUES COMPARISON..... 55
7. INTERACTION CAPABILITY..... 60

ACKNOWLEDGEMENT

The author wishes to express his appreciation for the efforts of Professor Gerald L. Barksdale, Jr., without whose guidance, encouragement and assistance this report would never have come about. Appreciation is also extended to Mrs. Cecelia Lupton for her support, understanding and patience.

I. INTRODUCTION

The basic security requirement in any system is to prevent unauthorized access or change of data while allowing authorized use necessary to accomplish the system's mission. Manual systems require the protection of data only. Automated systems introduce the added problem of protecting the process, both programs and hardware, that are used to store, access and change the data. This implies the necessity of adequate safeguards built into management, and hardware/software aspects of the system. Thus a decision must be made as to what is needed in the way of security. The processing and storing of sensitive information and preventing it from falling into the wrong hands is a technological problem requiring a comprehensive examination of both the type of information and the possible/probable threats that a data bank will be required to handle.

The objective of the current effort is to (1) review current and completed studies of the security and access limitation problem for automated systems; (2) analyze the data collected considering the differences in techniques as required by systems and users; (3) discuss techniques for data base security and access control applicable to a given system.

II. NATURE OF THE PROBLEM

A close examination of security requirements indicates that they are dependent on the specific type of threat posed to the system. There are three general classes of threats: unintentional, deliberate passive, and deliberate active.

Unintentional threats are those that arise from hardware and software failures and user errors which allow unauthorized but inadvertant access to files or programs.

Deliberate passive threats are caused by electromagnetic radiation from the computer hardware and communications equipment. Passive methods include wire-tapping and monitoring of electromagnetic emanations.

Deliberate active threats are from attempts to enter the system so as to obtain data from the files or to interfere with data files or the system. Examples of this type of threat are (1) using legitimate access to ask or obtain unauthorized access (browsing), (2) masquerading as a legitimate user, (3) using access to the system as support personnel (systems programmers, operator, hardware maintenance, management) to obtain data or create trap doors into the system, (4) tapping into remote terminals to receive "piggy back" entry with an authorized user, (5) between lines entry, and (6) cancellation of user's sign off signals to continue operation.

These threats are nearly the same for all systems, differing primarily in the degree which system design features allow exploitation. This potential for exploitation is created at each point where a user interacts with the system. Since the security requirements depend on the threat of exploitation and the threat of exploitation in turn depends on the particular system access point, the key to specifying the security requirements for a system lies in an examination of the systems accessibility.

III. SECURITY REQUIREMENTS COMMON TO ALL SYSTEMS

The overall safeguarding of information in a computer system, regardless of configuration, is achieved by a combination of protection features aimed at the different areas of leakage points. These areas are discussed and an overview of the vulnerability points are depicted in fig 1.

A. MAINTENANCE AND SUPPORT ACCESS

All systems have the requirement to allow access for maintenance of the system software and hardware. This "support access" must be provided for the system programmers, maintenance personnel, computer operators, and management personnel responsible for the system operation. It represents a potential means of deliberate active penetration and has been addressed in the literature on non-military systems as the area of most concern. Non-military systems lack the procedural security regulations established by law for military systems. [AMR 71]

The support access characteristics were similar in all systems. Access at the assembly/procedure oriented language level is needed to debug programs, maintain hardware, and establish system operating conditions. In all government systems, this support activity was conducted only at local terminals within the secure computer area, where common procedural techniques were relied on to limit access to cleared personnel.

The accessibility afforded to support personnel in commercial systems has received more attention in terms of the development of sophisticated automated security techniques. The primary reason is that military systems have developed strong procedural techniques (clearance procedures for personnel, security regulations with formal

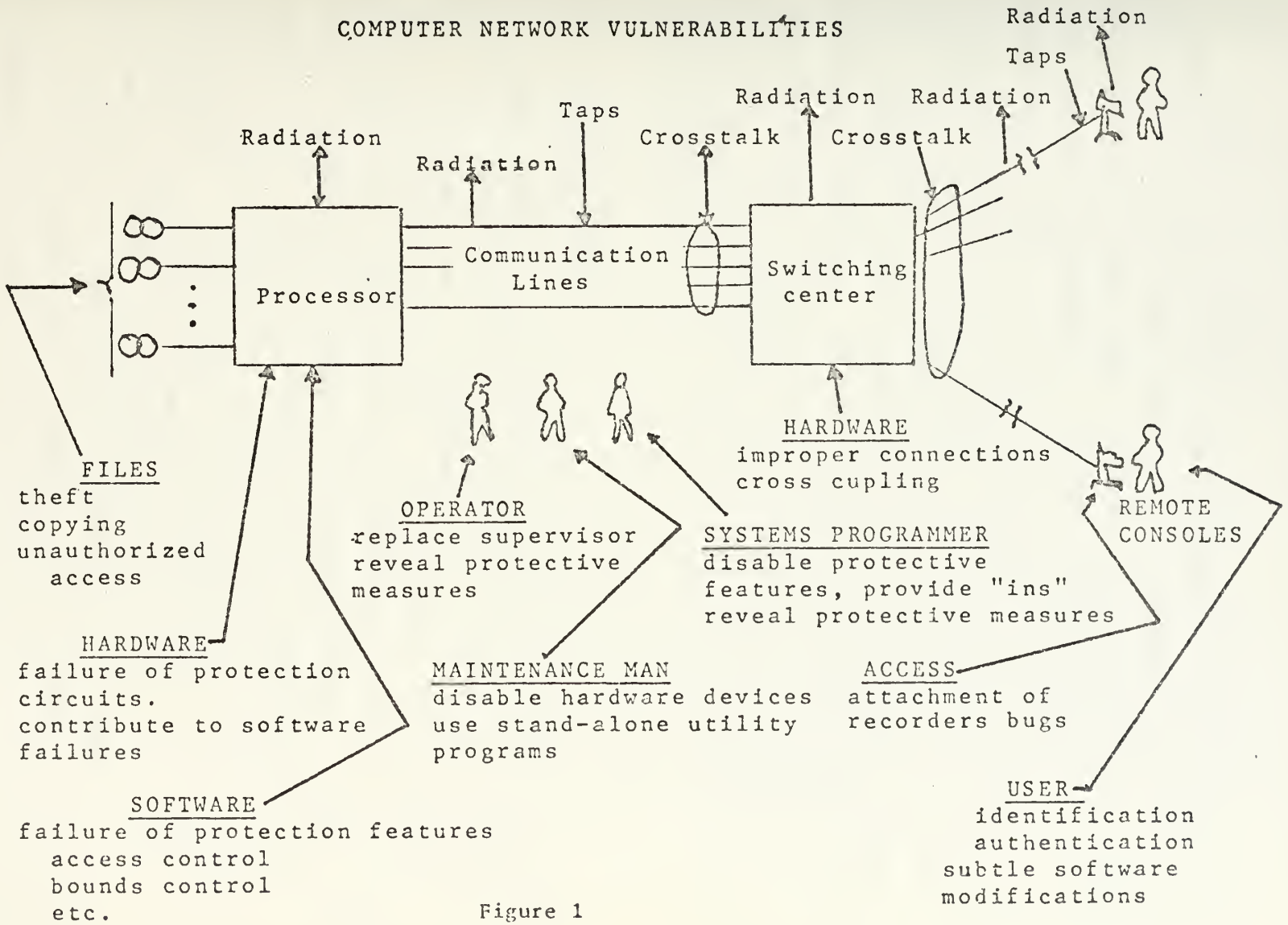

COMPUTER NETWORK VULNERABILITIES
Radiation
Taps
Radiation
Radiation
Taps
Crosstalk
Radiation
Crosstalk
Processor
Communication Lines
Switching center
FILES
theft
copying
unauthorized access
HARDWARE
failure of protection circuits.
contribute to software failures
SOFTWARE
failure of protection features
access control
bounds control
etc.
OPERATOR
replace supervisor
reveal protective measures
MAINTENANCE MAN
disable hardware devices
use stand-alone utility programs
HARDWARE
improper connections
cross cupling
SYSTEMS PROGRAMMER
disable protective features, provide "ins"
reveal protective measures
ACCESS
attachment of recorders bugs
REMOTE CONSOLES
USER
identification
authentication
subtle software modifications

Figure 1

legal penalties for infractions) that are not available for use in commercial systems.

Security requirements determined by the need for support access include (1) the ability to isolate access to programs and data to only those authorized to maintain the particular program/file, (2) the ability to effectively restrict maintenance personnel to the maintenance of specific software routines, (3) the need for procedures to insure that programs are completely debugged, (4) the need to audit files for unauthorized changes, (5) the ability to determine if equipment is operating properly before it is placed in the system, (6) the ability to detect and control changes to systems routines, (7) the ability to bound dumps of memory and peripheral storage, (8) the ability to determine that a program only performs the function for which it was designed, and (9) the ability to restrict access by internal control tables.

B. FAILURE ACCESS

The threat of compromise from the release of data or programs due to hardware or software failure is common to all systems and represents a potential means of unintentional penetration. Such failures can involve the coupling of information from one user with that of another user, rendering the files or programs unusable. They could result in defeat or circumvention of security measures, or unintentional change in security status of users files or terminals. Accidental disclosures may also occur by improper actions of machine operating or maintenance personnel without deliberate intent.

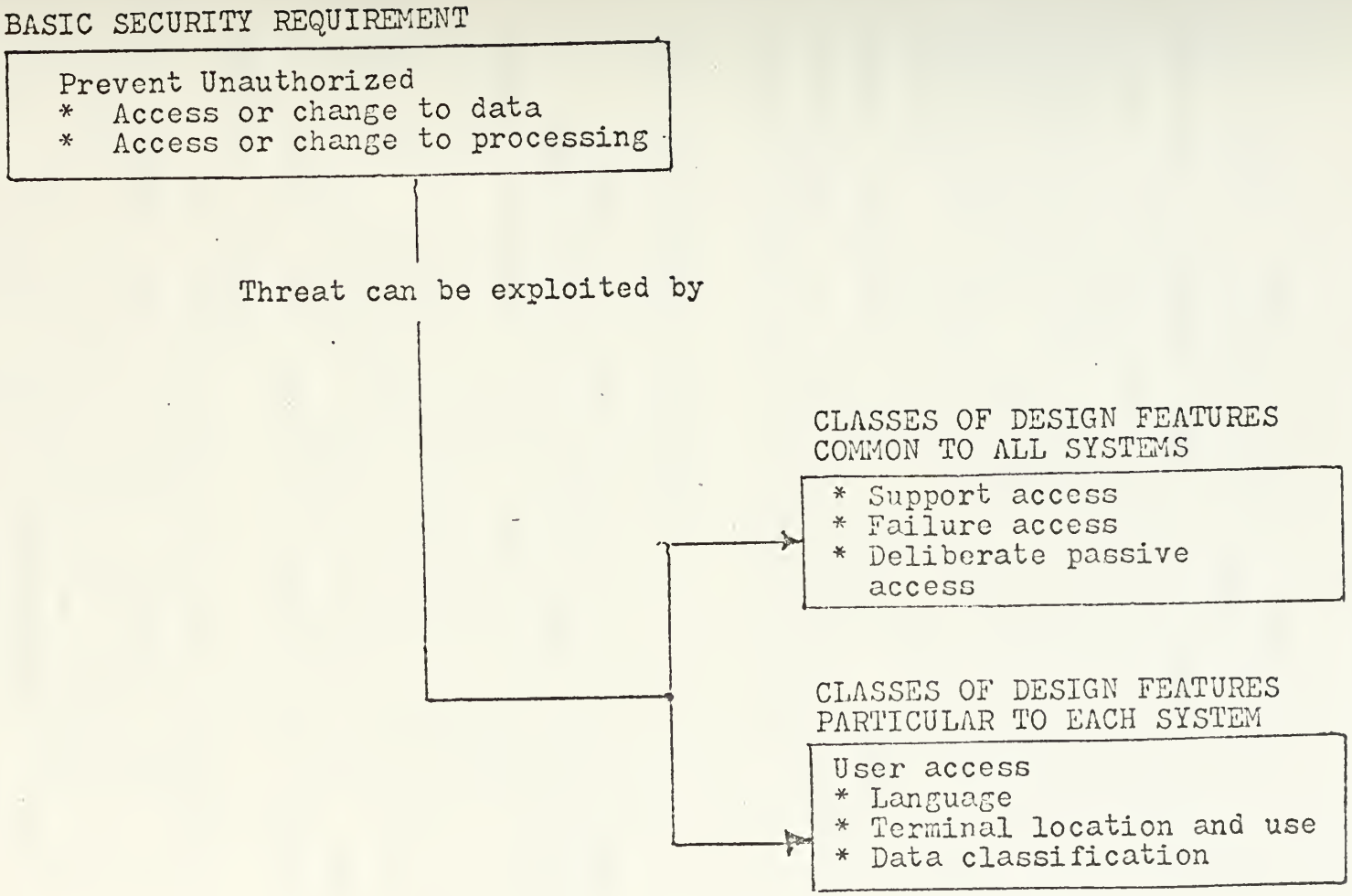
Security requirements determined by the need for failure control include the following: (1) the ability to trap to software error routines when parity errors are encountered, (2) the ability to prevent circumvention by software "bugs" of the partitioning technique that isolates data and

programs from unauthorized access, (3) the need to check out and certify program changes and equipment repairs to ensure that they are operating correctly, (4) the need to maintain the protection mechanism when a system error is encountered, (5) the need to recover from failure without revealing protected data and system tables, (6) the need to protect back-up files, and to certify that the appropriate backup file is loaded.

C. DELIBERATE PASSIVE ACCESS

Electromagnetic radiation, wiretapping, and "bugs" can be used on all systems if proper security techniques are not implemented. Electromagnetic radiations from computer equipment power lines and communications lines can be detected and decoded. Wiretapping into communications lines can be used to send and receive data. The planting of bugs or recording devices is possible if proper area security precautions are not used. Techniques that are applicable to this category of requirements are cryptographic data transformations and/or shielded lines for communication links, and maintenance of a properly secured area.

Security requirements determined by the need to prevent deliberate passive access include conventional red/black (classified lines & unclassified lines) isolation requirements as well as (1) the ability to encode/decode transmitted data so that it cannot be deciphered. (2) the need to certify that hardware either can not be, or has not been tampered with and, (3) the ability to make stored data unintelligible to direct dumping.



Classes of Design Features that Determine System Security Requirements
Figure 2

IV. SECURITY REQUIREMENTS PARTICULAR TO A GIVEN SYSTEM

Systems differ in their interface with the system user. The user of the system exercises the system for its functional purpose and is not concerned with its design, implementation, or maintenance. User accessibility to a system is defined by the type of system interface, the language capability offered, and the clearance of data and users provided. Different combinations of these imply increasingly sophisticated levels of access rights and hence different possibilities of penetration attempts. User access capability that directly relate to security requirements are language capability, terminal location and usage and, user and data clearance levels.

A. USER LANGUAGE CAPABILITY

The user interfaces with the automated system in either an off-line or an on-line mode. In an off-line mode he submits requests for data services to support personnel and receives as his output printed reports. This mode of operation is typical of closed shop batch systems. Their security requirements, in so far as they concern the user, differ from manual systems only by the addition of a requirement for security within the secure area of the computer facility.

In the on-line mode, the user is provided a capability to request data services directly from the computer equipment by means of some input (generally remote) device. His form of interaction with the computer can vary from rigid requests for predetermined fixed transaction input/output, to use of a free-form query language, and entry of actual computer programs in procedure oriented or assembly language. Increased security requirements are dictated as the level of capability to access data

increases.

Fixed transaction input/output allows the system designer to predetermine what will be the specific input and output allowed for a given user at a given terminal. The ability to enter and execute POL or assembly language programs places the user at almost the same capability as support personnel and could allow circumvention of security techniques implemented for fixed format or free form query capabilities. A higher level of security requirements is necessary to provide protection against this increased language capability. See figure 2.

B. TERMINAL LOCATION AND USAGE

The ability of a user to access or change data from remote terminals suggests penetration methods not possible in a system with only local terminals within a secure area. The communications lines must be protected and systems or user errors could allow release of data outside the secure area. Additionally, the vulnerability of a remote terminal secure area, especially in a tactical military system, is greater.

System security requirements are also influenced by the use of the terminals; that is, whether there is only one class of need-to-know at a given terminal or whether there are multiple classes of need-to-know at a given terminal. Multiple need-to-know at a given terminal requires that the system be able to identify the different user classes at a given terminal and provide protection against "browsing" and "masquerading".

C. DATA CLASSIFICATION

The security classification of data is an expression of the value of the information to national defense and hence the seriousness of its unauthorized access or change.

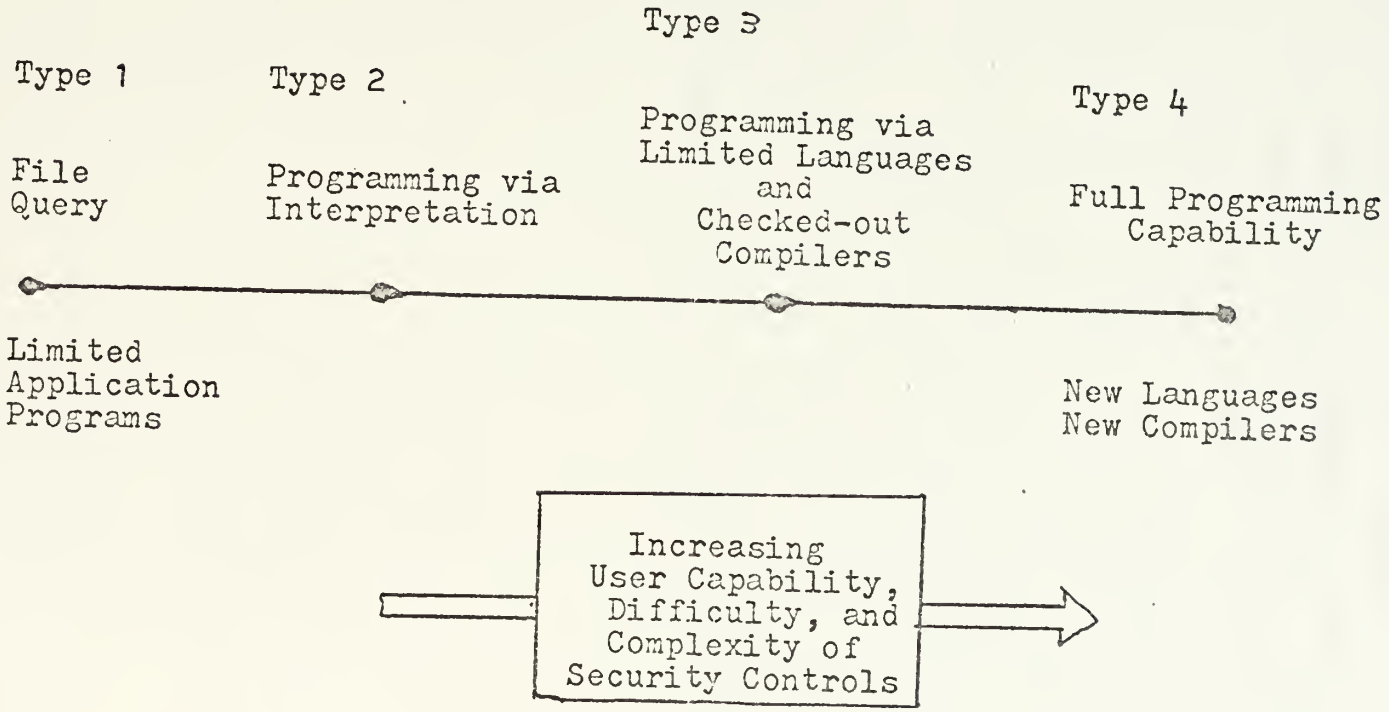

Type 1
File
Query
Type 2
Programming via
Interpretation
Type 3
Programming via
Limited Languages
and
Checked-out
Compilers
Type 4
Full Programming
Capability
Limited
Application
Programs
New Languages
New Compilers
Increasing
User Capability,
Difficulty, and
Complexity of
Security Controls

Figure 3

Systems which handle Top Secret data have a higher security requirement than those which handle data of lower classification. In a security requirement sense, if different levels of data classification exists, the security design problem increases since techniques must be implemented to isolate the different levels, provide the proper degree of security protection, and guard against unintentional or deliberate attempts to gain access to data at unauthorized security levels.

Three levels of security requirements exist depending upon the classification of data in the system and the level of clearance of the user: (1) the data classification is all one level (such as Secret) and all users are cleared to that level. (2) different data classification exists (Top Secret, Secret, Confidential) with users of different clearance levels, and (3) unclassified data exists with classified data and uncleared users are allowed to access the unclassified data. The security requirement increases as data classification and user clearance level increase in complexity.

V. COMPARISON BETWEEN HARDWARE, SOFTWARE, and PROCEDURAL TECHNIQUES

The comparison of hardware and software techniques identifies which security function can best be accomplished by each and whether combinations of these techniques are necessary to provide adequate protection. A comparison of manual versus automated systems procedures identifies the similarities between the two systems and the different approaches taken to perform the security function. Such a comparison provides a method to judge the relative value of automated techniques to achieve at least the same security level as manual systems. This section considers first the comparison of hardware and software techniques which can be used for security purposes and then considers the similarities and differences between automated and manual system security procedures.

Figures 4 and 5 provide a qualitative assessment of the relative merits and costs of the techniques discussed. These costs are divided into three areas. Costs for procedural techniques were not estimated.

Response time - the cost incurred by every message input to the system expressed as an effect on the length of time that a message response is delayed by the processing required for the technique in question.

Throughput - the cost incurred by the system expressed as the decrease in the amount of processing the system is able to accomplish in a given time period caused by the additional processing required for the technique in question.

Procurement - the cost associated with each technique expressed as the degree of expense involved in developing, maintaining, and servicing the technique in question.

The effect in each case is described as low, medium, or high, where low is taken to mean less than 5 percent

increment in the cost of the system, medium to mean between 5 and 10 percent, and high to mean more than 10 percent.

A. SOFTWARE TECHNIQUES

The software techniques surveyed are described and compared in terms of the general threat to which they apply. The techniques are categorized by the major functional routines of an on-line system.

1. User Interface

The user interface is the point at which the user becomes known to and interacts with the system. In a secure system, only known users can be permitted access. The proper identification of the user is necessary for accountability and, in a system that allows multiple need-to-know access at a terminal, to determine the access rights to be associated with the task that is initiated by a users input request.

a. User Security Clearance

The user security clearance is the assignment to each user of a code word indicating the highest classification level of data to which he has been authorized access. Generally, the code word consists of three bits, allowing for seven combinations, that are compared on a simple equality test against the security classification code of the data. The assignment and maintenance of codes are the responsibility of either the security officer, the data administrator, or support personnel. The legal pairs of user codes and data classificiation codes are maintained in most cases in a system table which can only be accesed in executive mode.

b. User Access Privileges

If it is necessary to link individual users to subsets of the available data or processes, then some type

SOFTWARE TECHNIQUE SUMMARY

TECHNIQUE	SECURITY THREAT	RELATIVE COST RESPONSE	THROUGHPUT	PROCUREMENT
User classification	Identify profiles	low	low	low
User access privileges	User accessing illegal data	low	low	low
Password	Not identifying users	low	low	low-to-med.
Password dialogue	Input of legal code by illegal person	medium	low	low-to-med.
Consecutive password list	Compromise identification of password	medium	medium	low-med.
Password transforms	Read other user passwords	low	low	medium
Digital checkpoint and hashing	loss of data	low-to-med.	medium	low
Terminal answerback	Piggyback terminals	low	medium	low
Terminal classification	Read out classified data at unclassified or lower classified terminal	low	low	low
Terminal character suppression	Inadvertant reading of system keys, or ID codes	medium	low	low

Figure 4

Automatic alarm disconnect	Remote terminal interloper	low	med.-high	medium
Privileged instructions	Accessing illegal tables	low	low	low
File classification code	Not distinguishing between several levels of classified data	low	low	low
File access lists	Unauthorized access	low	low	low
File access lists by file	As above, but more granular	low	low	high
File access profile	Remote user browsing in files	low	low	low
Data element classification	Mixed data classes	med.-high	medium	high
Data element profile	Multiple access privileges	med.-high	medium	high
File encryption - single key	Reading classified data without legal access	low	medium	low-med.
File encryption - multiple key	Reading classified data without legal access	low-med.	high	med.-high

Figure 4 cont.

Data edition number	Mutual interference	low	low-med.	medium
Block write-collision	Multiple concurrent users	low	med.-high	high
King structures	Modify, read or execute	low-med.	medium	high
Relocatable bootstrap	Read executive program by selected dumps	low	low	high
Redundant coding	Modifying executive routines	low-med.	high	high
Module dialogue	Accessing routines without privilege to do so	low	high	high
Program interpretation	Inputting illegal procedures or changes	med.-high	high	high
Centralized I/O control	Bypassing table protection of data	low-med.	medium	low
Error monitors	Failure in validity checks	low	medium	medium

Figure 4 cont.

Error interrupts	Invalid transmissions	low	low	low
Executive commands by access rights	Developing processes to access or change data	low	medium	medium
Boundary maps	Read/write outside of assigned area	low	low-med.	low
Memory access keys	Inadvertant overwrites	low	low-med.	low
Security monitor	Detect illegal entry attempts	low-med.	medium	med.-high
Document Log	Detect unauthorized input attempt	low	low-med	low
Erroneous attempts limit	Random input of combinations to obtain legal hit	low	med.-high	med.-high
Aggregate techniques for reports	Inference of higher level access	low-med.	low-med.	med.-high
Overwrite and memory erase	Discovering classified residue	medium	med.-high	low
Classified programs	Access to generations routines	low	medium	medium

Figure 4 cont.

Classified headers & trailers	Mixing classes of data	low	low	low
File log	undetected accesses	low	medium	low

Figure 4 cont.

of user profile must be developed either pointing to or specifically identifying the user access privileges. This profile may contain the identity or classification of the files available to the user, the manner in which the files can be accessed (i.e., read, write, process, modify or erase) the degree to which access is permitted, the specific terminals from which the user may operate, and the particular processes he may execute (named routines, standard jobs, or precompiled transactions). [Glasser 67] The user profiles themselves are maintained as a system file, normally resident on secondary storage because of its size. Because of its sensitivity, the data contents of the file are usually transformed.

c. Password

The password is the privileged identifier that a user must submit to obtain entry to the system. From a software standpoint, it is the only means of initially identifying a legal system user. Passwords may be required at log-in, at both log-in and log-out, or for every transaction executed. The more frequently the password is required, the less likely is the possibility that an illegal user will obtain entry but the more costly the user interface becomes in terms of its effect on the thruput and response time. Typed-in passwords range from 3 to 18 alphanumeric characters in existing systems, are either fixed or variable in length and may contain blanks. No data was available on the format of voiceprint or key-pattern passwords. Passwords remain unchanged in some systems, are changed periodically in others, and are changed at irregular intervals in one proposed design. [Weissman 69] The more frequently the password is changed, the higher are both the maintenance cost and the error rate. Passwords are either assigned, generated, or selected using some standard random number generator.

Although it has been shown that any password scheme can eventually be broken, the degree of difficulty of

doing so exceeds that of opening a 3-way 50-number safe-combination when the password exceeds five alphanumeric characters. Some systems attempt to detect password tinkering by assuming that a fixed number of consecutive illegal attempts (usually 2) from the same terminal is sufficient cause. Legal passwords are maintained in every case as a system table.

d. Password Dialogue

Since it is possible to eventually break any password scheme, several variations of the techniques have been suggested to obtain more foolproof identification of legal users. One such variant is to require the user to engage in a form of dialogue with the system after the initial password is validated. This dialogue requires the user to provide responses either unique to himself (his payroll number in one case; another password in another case; a user defined item of personal knowledge in a third case) or to perform some relatively simple algorithm on either a system-supplied random variable or some transitory quantity (time of day, date, etc); the system performs the same algorithm and checks the validity of the response. [Babcock 67] (e.g., system:"enter password", user: shazam, system:"OK. enter key", user: 3750094) Once again, the scheme is susceptible to penetration, but the level of difficulty has been raised significantly - at a cost in increased terminal response time and communication line loading. [Baran 64]

e. Consecutive Password List

Another variation of the basic password technique is to assign a list of legal passwords to each user. The system will accept only the next password on the list each time that the user enters his password. This makes it extremely difficlt to obtain a legal password through either passive deliberate penetration attempts or active tinkering with password combinations, but it also requires hard copy lists of legal passwords to be made

available to users and inevitably produces a greater number of erroneous entries by legal users. In spite of these drawbacks, consecutive passwords for each input has been accepted as an alternative to encrypted data links in one military system. [Weissman 69]

f. Password Transform

Since the list of legal passwords is considered to be extremely sensitive information, it is usually resident in core, and is frequently appended to program status blocks. Several systems have taken steps to prevent its being obtained either deliberately or accidentally by a readout from core. These steps involve implementing various transformation techniques on passwords received. Huffman encoding is used in one system; a simple transposition of digits is used in another; an algorithm to produce non-reversible inversions is implemented in a third. [Petersen 67]

2. Terminal Subsystem

Nearly all of the systems surveyed provided for on-line terminals and a significant part of the software in these systems is that associated with terminal characteristics. Less significant, however, are the software techniques implemented to account for security requirements arising from remote terminals. Terminals must be discretely identified to insure that data is transmitted to the correct location. Terminals at remote sites are susceptible to communication errors on transmission due to noise, may be easily expropriated for illegal use, and are subject frequently to public or semi-private display and the subsequent casual eavesdropping.

a. Error Correction Methods

The most obvious problem with terminals connected to a system through communication lines is the noise factor on the communication lines themselves. It

introduces the possibility that illegal values or erroneous addresses may be input in otherwise valid messages. Methods have been developed in many systems to reduce the effects of noise in transmission. They include hash totals (i.e., cumulative adds) of characters or bits in the message; parity check bits and longitudinal redundancy checks to detect garbled words; and retransmission to compare duplicate results.

b. Terminal Answerback

Since it is possible to piggyback illegal terminals onto legal circuits, particularly in dialed and switched network systems, methods have been developed to uniquely identify legal terminals. Often, the identity is established by comparing the expected terminal address to a hard-wired terminal identifier that automatically transmits (i.e., "answers back") an identification-key (20 alphanumeric characters in the system where this figure was published) with each input message or in response to a request code preceeding each output message. [CDC 66]

c. Terminal Profile

The classes of data and/or users that can be legally associated with a given terminal are defined in a terminal profile list. This list is an extension of the terminal address table maintained by the executive. It usually only describes the highest security classification of data that can be output to a given terminal. It may also include a list of specific transactions that can be executed from that terminal and/or a list of explicitly named users who may access through the terminal. [Weissman 67]

d. Terminal Character Suppression

Any on-line terminal that is used to input identification codes is susceptible to both casual and deliberate eavesdropping. There are two variations of a technique to reduce the vulnerability of input codes. In the case where hard-copy is used, the system strikes over the number of positions required for identifier codes each

time such a code is expected from the user. This provides a marginal degree of protection. In the other case, a code is transmitted to the terminal to suspend printing or display-images on the text-line in which an identifier code is expected by the system.

e. Automatic Alarm and Disconnect

If the system is able to detect that a terminal or a terminal connection is being used for (attempted) illegal input, it is necessary to provide an alarm to alert the control group and to isolate the suspect terminal from the system. Obviously, the alert should include the terminal address. It could also include the nature of the attempted input. Since it may be advantageous not to alert the interloper that his prescence has been detected, the isolation of the suspect terminal in one suggested plan would still permit it to remain linked to the system by engaging the user in a series of questions and delays. In most cases, however, the terminal is disconnected and/or the keyboard is locked to prevent further communication. Bringing the terminal on-line again usually requires that the security officer or control group input a special identifier code.

3. Executive/Monitor

The heart of any multi-programming system is the executive control routine. It is the most complex, sophisticated, and important component of the software. By its very nature, it is perhaps the most difficult to penetrate but then, it is undoubtedly the most rewarding. In this area in particular, procedural techniques must be relied upon. It is impossible to prevent support personnel from leaving "trapdoors" or potential entry points in the software and the need for integrity of and confidence in support personnel is paramount. Many of the techniques developed in this area also require parallel hardware

features.

a. Privileged Instructions

Given that the hardware has a master/user mode capability, the set of instructions that can be executed in the master mode are regarded as privilegded instructions. Since they are intimately involved with system control,(e.g., the setting and resetting of bounds registers, the initiation of channel commands, the loading of read/write address registers, the deciphering of internal and external interrupts) they have an immediate application to security requirements. They should be used sparingly and should be concentrated in a few easily associated routines. The routines should operate in priviledged mode as briefly as possible, branching to user mode to perform the function initiated. Dispersing priviledged instructions in many executive routines simply improves the chances for trapdoors and illegal circumvention.

b. Relocatable Bootstrap

If it were possible to bypass protection keys and to gain access to areas of memory normally reserved for the executive and its tables, then it would also be possible to read any of the access lists and authority tables controlled by the executive. One technique suggested to reduce the likelihood of this occurring is to perform bootstrap loading of executive routines from a changing key address. In this manner, executive routines and tables no longer have absolute locations relative to each other and to the user partitions, and only haphazard location of the secured routines would be possible. Because of its potential effect on the efficiency of the system, the technique has only been discussed. [CDC 66]

c. Redundant Coding

Since it is possible to modify code prior to loading, it has been suggested that key routines exist as multiple, discrete copies, and that requests for the services of these routines be executed in parallel by each

copy. The results can then be compared, including number of instructions executed. The effective cost of this approach is high even in a multiprocessing system, but it further insures that key security routines cannot be modified or executed without detection. [Molho 70]

d. Module Dialogue

In any calling sequence, the parameters passed between modules are usually specified as a part of the standard call macro. It has been suggested that this be modified some-what in those cases where it is feared that an interloper may substitute his own code for a system routine. At random points in the routine in question, private call parameters known only to the programmer responsible for that routine can be inserted. The routine called (or calling) is also prepared to expect the interpersed dialogue words. Since these would be difficult to detect in absolute code, it would raise considerably the level of difficulty associated with making such code substitutions.

e. Program Interpretation

Since it is difficult to detect subtle changes in absolute code, it has been suggested that programs be loaded through an interpreter at all times. If the interpreter includes some kind of code cptimizer, each version cf a program in its absolute code form would be slightly different than the preceding one. In this way, not only would it be difficult for a penetrator to modify or decipher program routines (except as a one-time event), but it would also be difficult for the programmer himself to take advantage of fixed relationships in his program that might permit the introduction of trap-doors.

f. Centralized I/O Control

This is a fairly common technique employed by most third-generation systems. It separates application programs from direct address references to I/O devices and instead requires them to submit macro commands that deal with the device as a logical, virtual, or relative extension

of memory. The executive then generates and performs the physical I/O commands and thereby is able to maintain control over boundary establishment and limited units of allocation. Without the equivalent hardware capability to trap to monitor mode whenever privileged instructions are attempted, this approach cannot be validated.

g. Error Monitors

In a system with many users, the cost of maintaining security can increase significantly if the user error-rate is high. This technique is intended to maintain a rating of the capability of individual users to perform the procedures associated with inputting valid transactions. If their error-rate increases beyond a predetermined level, then their priority in the system is decreased. The cost of maintaining this scheme is quite high however, since it requires some corresponding method to re-evaluate and to certify the user's capability.

h. Error Interrupts

Any attempt to perform an illegal operation, to address some location outside of assigned boundaries, to input erroneous data, etc., should be the cause of an error interrupt. The routines to handle such interrupts can attempt to correct the error and resubmit the request, abort or suspend the user in question, alarm control authorities, or regard the error as acceptable, flag it, and continue processing. Once it is determined in monitor mode what the interrupt is, any further processing to deal with it should be performed in user mode to reduce the occasions for illegal execution of privileged instructions.

i. Executive Commands By Access Rights

This technique associates a category code with all executive command routines and restricts their direct use to only the subset of users cleared to the equivalent category of access.

j. Boundary Maps

Boundary maps are the legal units of allocation

assigned to given users. They represent the direct input to base and limit registers that determine the domain in which a user can be active. Boundary maps in most cases are stored with the user in question and represent a potential means to illegally extend the accessibility afforded a given function should they be modified. [CDC 66]

k. Memory Access Keys

In a page or segment-oriented system, there are usually lock registers associated with each physical page in memory. When a user is assigned to memory, his identifier is used to generate a unique key that is loaded into all of the page registers assigned to the particular user. An address reference to the protected pages cannot be made unless it contains the appropriate key-pattern in its own key register. Obviously, selected executive routines must have a universal key. Setting and access to the key registers should be a privileged function. [IBM 67]

l. Security Monitor

The security monitor is a technique that attempts to certify the validity of the various protection mechanisms in a system. At its simplest, it consists of a set of on-line diagnostic routines that exercise the various hardware components in a configuration, expecting a valid operation tc produce a pre-designated result. In a more complex form, it attempts to deliberately execute illegal hardware or software operations and then determines whether or not the responsible protection mechanism has successfully intercepted and handled the illegal attempt; this version can have a significant effect on system thruput and, therefore, requires a careful consideration of what are acceptable and expected failure levels. [Molho 70]

4. File Handler

The data available in any system is the reward for penetrating the system. The data available in an automated

system significantly increases the potential reward because of the large amount, the anonymity of access, and the difficulty of detection. Data protection is traditionally obtained by assigning responsibility and limiting access. Techniques to accomplish both are available in automated systems. At least the same level of protection can be obtained in an automated system as in a manual system.

a. File Classification Code

This technique is commonly employed in most systems that deal with formally classified data. It simply involves assigning one of the categories of classification to each data file and then either assigning the file only to jobs or individuals of equal or higher clearance, or, in the case of shared files, releasing data from the file only to users of equal or higher clearance. A somewhat adventuresome extension of this technique is to attempt to automatically assign classification levels to new files. In one system, this is done by using the highest classification from contributing files. [Weissman 69] In a proposed scheme, it is done by doing a key-word count and weighing the file in accordance with the number of key-words encountered. [Daley 65] In neither case was it shown to be statistically more or less effective than manual classification, except in the marginal area between unclassified and confidential.

b. File Access Lists By File

An extension of the file classification code is the assignment of specific authority lists to each file. These lists describe the original creator (or owner) of the file, other individuals, group, terminal, etc., who can share the file, usually the manner in which they can access the file (read, write, modify, execute, or erase), and the degree to which access is permitted. At the file level only, it corresponds to the cataloguing function of most third-generation systems. [Glasser 67]

c. File Access Lists By Level

An extension of the previous technique that permits it to be used in a more flexible environment than that described by formal discrete files is to assign access lists to levels of files, or to individual modes in data sets. This is especially useful if the files consist of programs arranged in some kind of hierarchy from common (free) utility routines to machine-oriented (owned) system routines. The cost of maintenance, particularly the determination of access privileges is quite high. [Babcock]

d. File Access Profile

If a number of users with different need-to-know interact with a shared set of data, it is necessary to distinguish the data rights of each user. This is accomplished by assigning to each file descriptor a profile word that contains a set of flag-bits, each flag-bit representing a unique need-to-know identifer. In the systems utilizing this technique, separate profile words are assigned for read and update access. There is associated with each user's profile an equivalent word with the need-to-know flag assignment according to his requirement for data. A one-to-one correspondence between user and file profiles at each flag position is required before access is permitted. [Bingham 65]

e. Data Element Classification Code

This is identical to "file classification code" except that each data element in the file is separately classified. The system can then handle files with mixed classes of data. This feature greatly reduces the redundancy associated with file processing since it permits the grouping of data by functional purpose and utilization rather than by classification. However, it raises both the cost of creating files and the cost of assigning andmaintaining classification categories. [Weissman 69]

f. Data Element Profile

This technique is identical to "file access profile" except that the system can now discriminate among

need-to-know at a finer level of detail. It is usually implemeted by assigning an update word and an access word to each data element descriptor; these words have particular bit settings according to their class. Users with matching need-to-know patterns are permitted access to the data element. The bit-patterns are combined to form a composite need-to-know profile for each data request. In a few instances, the data element profiles also contains legal values of a data element that are accessible by a given class of users.

g. File Encryption, Single Key

Techniques for encrypting data have been suggested for use in file handling systems. In most cases, these are variations of cryptographic techniques applied to communication transmission, and consist of applying a single key to all records in the file. However, because of the large number of records in most data files and because of the rather consistant pattern of field occurrences, this type of file encryption only provides a marginal increase in protection. Depending on the amount of character manipulation in the crypto technique, CPU thruput cost can be quite high. [skatrud 69]

h. File Encryption, Multiple Key

A variation of the proceeding technique that reduces the possibility of deciphering is to use a different key (either a cascading or random number sequence) for each record or for certain number of records. This breaks the consistency of the encoded data and does not significantly affect the cost of encoding/decoding process. [Van Tassel 69]

i. Data Edition Number

In a system where multiple users are concurrently updating a set of shared data files, it is necessary to prevent one update sequence from intruding on another. A suggested tecnhique is to assign an edition number to every record in the data base. Each user contains

the latest version of this edition number in its associated data buffer. When a retrieved record is to be written back into the data base, the file handler checks the user's edition-number against the data base edition number and only permits the update if the edition numbers are the same. The file handler also updates the edition number. [Corbato 65]

j. Block-Write Collision

This technique addresses itself to the same problem as above, but does not attempt to control the interaction at the record level. Instead of an edition number a block-busy flag is assigned to each file segment. When a segment is retrieved for update, the block-busy flag is set, as are all antecedent blocks in the structure (or only the highest level block if the entry point is always top-down through the same index). The busy-flag is left on until the user has indicated completion of the update and the file-handler has modified the affected index blocks. [Babcock 67]

k. Ring Structures

Ring structures are a combination of logical layers, or rings of data grouped by sensitivity, and identifiers associated with each user that describes the equivalent sensitivity of the user. It is permissable for a user to access and/or execute any data or routine in its own ring. When a call is made to a segment in another ring, the system traps to a gate controller, which determines if the called ring is more or less sensitive. If the sensitivity is less, then the call is linked to the ring in question. If it is greater, then a check is made of the access list associated with the requested segment. This list identifies legal users (or classes or users), and indicates the type of access and the particular entry point at which they may use the requested segment. The gate controller then establishes the required linkages. To prevent repetitive calls to the gate controller, upper and lower bounds can be assigned to each type of access for any user; requests to any rings

within those bounds are automatic and equivalent to operating within the ring of the requesting user. The system that implemented this technique had special hardware registers to check the ring-brackets of segment requests. [Glasser 67]

5. Others

Varicus other software techniques were enccuntered during the data collection phase that do not conveniently fit into the preceeding categories. For the most part, these techniques have to do with using the automated system to simplify or extend some of the procedural requirements in a secured system.

a. Document Log

Some systems automatically maintain an accountability log making an entry each time that a classified report is related to a user. This log includes the date and time of the original request, the parameters specifying the report extraction criteria, and the terminal and user identification. The security log is available only to an identified security officer.

b. Erroneous Attempts Limit

This technique is applied at several intersection points between a user request and system function. Since a potential interloper can tinker with legality checks at any one of these points, it is necessary to set some limit on the number of consecutive illegal inputs that will be accepted from any user. Some type of on-line monitoring is required to record or link the sequence of requests.

c. Aggregate Techniques For Reports

A serious problem in on-line system is the possibility that even though a user cleared to a low level of access can only access data legally classified at or below his level, the aggregation of all data accessed can

provide the basis for interpretive conclusions about higher classified information. Techniques have been suggested that would combine the access profiles (data element profile) and all data elements contained in the report into a new profile that would yield a restricted classification of the report on a need-to-know basis. However, this cannot prevent inferences from data, and more work is needed in this area to determine if some kind of weighting of information content might be possible. [Feige 69]

d. Overwrite And Memory Erase

Any magnetic recording medium retains an electromagnetic image of the recorded data for some time after the initial impression. This residue can be read directly, albeit inadvertently, if access to the area is obtained or picked up through passive deliberate penetration attempts. Since both primary and secondary storage in most on-line multiuser systems is considered to be virtual memory, it is entirely possible that an area in which classified data had been stored and processed could be reassigned to a user having a lower classification level. To prevent this, methods have been developed to overwrite primary memory by cascading or leapfrogging thru the area and writing a system constant (usually zeros) after the memory space is deallocated. The confidence in this technique is increased if it is procedurally established that every user routine fills its scratch area with a different constant. In only a few systems is the same approach used for secondary storage, since the time required to overwrite deallocated file space on a peripheral device, particularly one with a single read-write head, can be considerable. If a centralized data manager is used by all system users for handling data files, it is conceivable that reallocated space can safely be maintained as "dirty" storage because it is not logically valid to read empty file space.

e. Classified Programs

This technique is used in batch oriented systems where, to the system, a user consists of a set of programs and their associated data files. Since the programs are designed to suit this single set of data, they take on the classification of the data and can only be called or modified by job control statements input with the proper classification leader.

f. Classification Headers and Trailers On Hardcopy and Displays

This technique is an extension of the current procedural technique of stamping at the top and bottom of every classified page of a report the classification level. It is usually a parameter option in the report generation routine.

g. File Log

The file log is an extension of the security log in which every reference to classified data is logged. It can include the previous data image if the reference causes a change. It also usually includes the terminal, user, time, data, and data parameters associated with the reference.

B. HARDWARE TECHNIQUES

Many of the hardware techniques required for security purposes have been implemented in third generation commercial computer systems and in some military computer systems. The surveyed techniques are assigned to categories that correspond to the major devices and components in an on-line system.

1. Central Processing Unit (CPU)

Security related techniques in the CPU provide control of the logical processes to access and change data. Techniques that isolate and control the operation of

HARDWARE TECHNIQUE SUMMARY

TECHNIQUE	SECURITY THREAT	RELATIVE COST		
		RESPONSE TIME	THRUPUT	PROCUREMENT
Processor Mode	Use of instructions which defeat software controls	low	Low	Medium
Core memory Bounding	Addressing into blocks outside user program area	Low	Low	Medium
Process Control Register	Attempts to execute unauthorized read/write executive instructions	Low	Low	Low
I/O Control Registers and Mask Registers	Wrong channel I/O	Low	Low	Medium
Parity Flag Bit Logic	Hardware error	Low	Low	Low
Security Control Flag Bits	Classified data released to unclassified terminal	Low	Low	High-extra bits in each word for security
Code Redundancy	Hardware error	Low	Low	Small to medium depending upon application

Figure 5

Key Redundant Registers and Logic	Hardware error	Low	Low	Low to medium
Parity Logic	Hardware error	Low	Low	Medium (extra bit for each word required)
Key Word Register	Attempts to execute unauthorized read/write execute and to access outside user core area	Low	Low	Low
Read Only Memory	Unauthorized alteration of programs or data	Low	Low	Medium
Dedicated Memory	Unauthorized access or alteration to programs or data	Low	Low	Medium to high
Memory Block Erase	Access to residue remaining in memory block	Medium	Medium	Low
Associative Memory	Access to residue, unauthorized access to data programs, data destruction	Low	Low	High
Memory Partitioning ports	Unauthorized access to special programs or data	Low	Low	Low to medium

Figure 5 cont.

I/O Bounds Control	Access to unauthorized main memory block	Low	Low	Low
Unit Address Register	Misrouting of data	Low	Low	Low
IOCP Parity Check	Hardware error	Low	Low	Low
I/O Channel/No. Character Check	Misrouting of data	Low	Low	Medium
I/O Security Level Register	Release of classified data on unauthorized channel	Low	Low	Low (High in memory requirement)
Channel Number Check Logic	Misrouting of data	Low	Low	Low
I/O Answer Back Check	Misrouting of data	Low to medium	Low	Low
I/O Memory Erase	Access to residue remaining in memory block	Low	Medium	Low
I/O Code Redundancy	Hardware error	Low	Low	Low to medium

Figure 5 cont.

Read Only Lock	Unauthorized change to programs or data	Low	Low	Low
Record Address Check	Misrouting of data	Low	Low	Low
Data Parity Check	Hardware error	Low	Low	Low
Check Sum Logic	Unauthorized change to data	Low	Low	Low
Cryptographic Devices	Access to data by electromagnetic radiation or wire tapping	Low	Low	Low
Hang Up and Dial	Misrouting of data	Low to medium	Low	Low to medium
Key Pattern Generator	Unauthorized access to programs or data	Low	Low	Medium to high
Combination Lock or Lock and Key	Unauthorized access to equipment	Low	Low	Low
Dual Hardware Access	Unauthorized access to equipment	Low	Low	Low
Data Destruction Techniques	Area seizure	Low	Low	Medium to high

Figure 5 cont.

programs and access to data, provide for recovery from hardware failure, and allow centralized error checking of programs and data access are also applicable to the security of the automated system.

a. Processor Mode, Privileged Instruction Set

Present third generation computer systems have implemented multiple modes of operations differing in the ability to process available instructions and in memory access restrictions. Typically, the system may operate in one of two modes: the control (executive mode) or the user mode. The processor will not execute a privileged instruction unless a processor mode register is set to the control mode. In the user mode, privileged instructions cannot be executed and memory accesses are restricted to those which were assigned while the processor was in the control mode. In the control mode, all instructions can be executed and all memory accessed. Should a privileged instruction occur in a user program or a memory access be attempted outside the allocated area, an interrupt returns control to the executive program. User programmed entry into the control mode is possible only by use of an executive request or monitor call instruction. Programmed exit to user mode from control mode is accomplished by executing a return-to-user mode privileged instruction. [Babcock 67]

b. Core Memory Bounding (BAR, Lock and Key, Paging)

Three major hardware techniques are used to limit core memory access of user programs to a bounded or allocated area established by the executive program. The base address register (BAR) containing upper and lower limits of allowable core memory access is used to insure that after indexing or indirect memory addressing, the hardware memory address is within the bounds of core memory assigned by the executive program. The lock and key memory bounds technique is implemented by the executive assignment of a key word to user programs and to memory areas that

defines the authorized core access area for that user program. The key word is automatically checked in the core memory before any access is allowed. The third method, paging, uses the key word designated by the executive as part of the hardware memory address and access is physically impossible outside the bounded core memory area. The techniques vary in terms of hardware cost: The base address register the lowest; lock and key, intermediate; and paging the highest.

c. Process Control Register (Read/Write/Execute)

The control of the right to read, write, or execute data has been implemented utilizing the basic method used for memory bounding. Flag bits are used in the associated memory bounds register to indicate the rights of the user program to read and/or write into core and to execute program instructions in a given memory area.

d. I/O Control Registers and Mask Register

The loading of I/O control registers and I/O mask registers allows the centralization of all input/output to executive control which is essential to the effective isolation of user programs and data. The mask register provides an effective means of controlling different types of interrupts including those associated with inadvertent or deliberate attempts of user programs to perform unauthorized actions.

e. Parity

Parity, generated for the transmission of data, is checked by receiving units. A single parity bit detects any single or odd number of bit errors in the word (character or bit group) in which it is included. It is used in most third generation computers to provide a method to detect hardware errors in all parts of the computer system. Detection of a parity error causes an interrupt to the executive mode. [Molho 70]

f. Security Control Flag Bits

Use of flag bits in programs and data for the

purpose of indicating security level is a technique which has been suggested but not implemented in any of the systems surveyed. The flag bits in each data word would indicate the security level of the data and the class of user who could read or write the data word. The flag bits in program instructions would govern execution of the program. Both hardware and/or software control has been suggested to interpret the flag bits. The high cost in additional memory to store these extra bits has been cited as one of the reasons why the technique has not been inplemented. Some of the more advanced systems have implemented the use of flag bits with software interpretation at the control word level rather than the individual word level.

g. Code Redundancy

The use of extra bits to provide code redundancy to enhance the capability to correct errors or to better identify errors has been suggested but not implemented for key control instructions. The mode control register and memory bounds register have been suggested as areas where code redundancy should be used. [Molho 70]

h. Redundant Key Registers and Logic

The technique of using redundant registers for mode control and I/O channel control has been suggested but not implemented. The use of multiple registers that would cause an error interrupt if they did not agree would insure the proper functioning of these key controls.

2. Main Memory Module

The control of access to the main memory module is necessary for proper security control in an automated system. All programs and data that will be accessed must ultimately reside in the main memory. Control of main memory areas in some systems is accomplished by the CPU, in others by a combination of CPU and memory circuits. Since all data and programs are read from the memory, the

integrity of data in the memory is paramount. Hardware failures must be detected to prevent the possibility of CPU execution of changed code with unpredictable results.

a. Parity Logic

The memory unit checks and/or generates parity bits for storage in order that odd bit failures can be detected. A parity failure causes an interrupt to the executive mode in the CPU.

b. Key Word Register

In lock and key memory protect systems, the memory compares the key flags of the access request with those set by the executive in the key word register. This prevents unauthorized read/write execute access to memory for data or programs.

c. Read Only Memory

Read only memory is used in some systems for key control programs to provide protection against unauthorized change to programs or data. The higher cost of such memories has limited their use.

d. Dedicated Memory

The use of separate memories for different classes of users has been suggested to provide security of classified data. Such a system requires control by an executive program. The physical separation of data into separate dedicated memories also requires correlation software techniques to code data. Plug-in dedicated memories for special programs and data such as security access lists and security monitor programs have been proposed but not implemented.

e. Memory Block Erase

A special instruction and associated hardware to clear a specified block of memory has been suggested in order to clear residue from a task upon completion of that task. This procedure would insure that classified data is destroyed before memory is reallocated. This prevents core dump instructions at the beginning of the new user program

from outputting the previous user's data.

f. Associative Memory

An associative memory allows the retrieval of data or programs based upon a code match rather than a hardware address. It has been suggested as a technique that could be used to retrieve data or programs based upon a code designation of the data or program class. Large associative memories have not been implemented in systems because of their high cost.

g. Memory Partitioning Ports

The use of special dedicated ports or paths into dedicated blocks of memory has been suggested as a method to isolate special classes of data. The special ports could only be accessed by a special set of priveleged instructions.

3. I/O Control Processor

The input/output control processor(IOCP) provides the hardware interface with mass memory (disc, drums, tapes, card reader/punch) and with the systems user (printer, display, communications lines, etc.). Data is transferred between those external units through the IOCP to the CPU and main memory. The IOCP features which are particularly important to data security are the registers and logic that route the data between the proper external device and the proper main memory core block and CPU.

a. I/O Bounds Control

The CPU provides the starting address and either the word count or ending address for any data transfer between external devices and main memory. The IOCP, through bounds control, insures that the data is transferred to the allocated block of memory. Each address is automatically checked to insure that it is within the address bounds. Any address outside the address block causes an error interrupt to the executive mode.

b. Unit Address Register

The unit address for outputs of data is furnished to the IOCP by the CPU. Unit select gates connect the appropriate unit buffer in the IOCP based upon the contents of the unit address register. For data input, demand queueing is processed by the CPU which furnished the IOCP with a control word defining the allocated memory block for the input.

c. IOCP Parity Check

Inter-equipment address, data transfers and unit address control word parity checks provide the capability of detecting single failures and preventing the misrouting of data.

d. I/O Channel/Number Character Check

Logic has been suggested but not implemented to provide a means to identify misrouting of data in the IOCP. Before any data is released to a channel, the channel number terminal device would be checked by the CPU. A character count register would be set to allow transmission of a specified number of characters and decremented to zero as characters are transmitted. At zero count, the channel number would again be checked and the character count register reset. This procedure insures that data is being input/output on the correct channel and, in case of malfunction, limits the amount of data released.

e. I/O Security Level Register

This I/O register would check a record control security code word against a channel security level code. If the classification level of the control word was higher than the channel level, an interrupt would be generated. This channel security level check has been suggested but not implemented.

f. Channel Number Check Logic

This technique requires that the channel control word from the CPU to the IOCP be transmitted twice and matched by dual registers in the IOCP before data is

transmitted on the channel. A mismatch causes an error interrupt. The technique has been suggested but not implemented.

g. I/O Answerback Check

This suggested technique utilizes hardware which requires answer back identifying the receiving terminal unit before allowing any transmission to the terminal unit. The answer back terminal unit identification is checked for match against the original contrcl word in the IOCP before data is transmitted.

h. I/O Memory Eraser

The IOCP provides the capability to clear a block of memcry of residue from a previous use of the space. A control word from the CPU specifies the main memory block address. The IOCP then cycles through the block addresses transmitting all zeros to the block address. At the end of the block, the IOCP generates an interrupt to the CPU that identifies the block as being cleared. This technique could be used in CPU limited systems instead of a software routine.

i. I/O Code Redundancy

Additional bits over those logically required could be used for terminal addressing. This would provide the capability of error detection and error correction. Additional hardware error detection and correction hardware would be required.

4. Direct Access Memory Controller

The majority of the data and programs are stored on direct access memory (disc, drum, tape) and transferred to the main memory when required for processing. Physical and electrical control of access to these devices is necessary to insure security. Electrical access to the devices is through their controllers which provide for record address location and read/write execution on the device.

a. Read Only Lock

Logic and switches on the controller provide the capability to allow read only access on specified tracks of disc and drum. n specified tracks of disc and drum. These switches are set to read only or read/write at system setup time. tape drivers can also be set to read only by write disable switches.

b. Record Address Check

The controller checks for parity of each word as it is read from the device. A parity error generates an error interrupt to the IOCP.

c. Check Sum Logic

The controller counts the bits in a given record and checks this total against a total entered at the beginning cr end of record. If the total does not agree, an error interrupt is generated. This technique allows the detection of unauthorized change to records.

5. Remote and Local User Terminals

The computer system terminals are the means used for communication between the automated system and the user. Access to the terminals and user capabilities allowed at the terminals are the key security control features. The hardware security techniques identified at the terminals provide means to limit access and control user capabilities.

a. Cryptographic Devices

Cryptographic devices are used to automatically encode and decode data on communications channels. The techniques used in the devices are highly classified and require a special engineering discipline. Therefore, it becomes a cost-effective decision as to the use of cryptographic hardware devices. For communication channels, cryptographic techniques are the only known practical method to prevent access to data by radition or wire tapping. [Van Tassel 69]

b. Hang Up and Dial

This technique provides logic that transmits a request from the terminal for computer services. The computer then requests a verification of the request from the terminal. An identification code is automatically sent that confirms the terminal request. Separate lines have been suggested for the transmission of the two identification requests. In some systems, a telephone confirmation by computer support personnel is used for the verification of the on-line terminal request. [Petersen 67]

c. Key Pattern Generators

Several techniques to identify individual users have been suggested. Identification card readers are used by a few systems. Voice, fingerprint, and combination lock code generators have been suggested but not implemented for the generation of individual key patterns. The key patterns are transmitted to the computer system where access rights to data and programs are authorized on the basis of the key pattern comparison.

6. General Techniques

Some of the hardware techniques apply to more than one of the subsystems of the computer. They are described in this section.

a. Combination Lock or Lock And Key

The physical securing of key parts of the computer system by combination lock or lock and key has been suggested as a method to limit access to critical circuits. The circuits suggested for this protection are the power circuit at terminals, read only switches on mass memory devices, the cabinets that contain the IOCP, CPU, core memories, and dedicated plug in memories.

b. Dual Hardware Access

This technique would require the simultaneous insertion of keys by more than one person to gain access to

key hardware components.

c. Data Destruction Techniques

The problem of quickly destroying classified data in cases of seizure of a computer area or a remote terminal is a security problem in both automated and manual systems. The problem is not an easy one since large volumes of data must be destroyed in a short time. The sequential writing of a random number stream on to data files does not prevent residual effects on magnetic storage devices, but would make data recovery much more difficult. Physical destruction of devices, depending upon the level of destruction, could destroy the data involved. The degaussing of the mass storage devices is a possibility but could require too much time or unreasonable power levels. It is felt that further study is needed to identify reasonable and practical methods to provide for protection from the threat of area seizure.

C. HARDWARE AND SOFTWARE COMPARISONS

Effective on-line control to prevent one user's programs and data from being accessed or changed by other user's programs can be achieved by hardware techniques. The use of a processor mode, priviledged instruction set, and memory bounds provides the tools for effective isolation of programs and data. The effective on-line control of files or data from unauthorized access can best be achieved by software techniques. The use of user profile tables (contains user code identification, program rights and clearance level) provides an effective means to control access to programs and data. Software programs can also provide security monitoring and security logging of all access or changes to data and programs. [Carrol 71]

Either software or hardware techniques can be used for communications channel coding of data, the clearing of residue data from main memory blocks, terminal user

identification, detection of unauthorized change to data on direct access memory, and proper I/O routing.

Software and hardware techniques both are considered necessary for recovery from software failures, and if required, the effective isolation of support user's access to data and programs. The isolation of support personnel from access to data and programs is the most difficult automated security technique to implement. The hardware techniques required consist of processor mode and privileged instruction set and could include the use of dedicated memory. Software techniques for isolation that have been suggested but not implemented include relocative boctstrap, redundant coding, module dialogue and program interpretation.

Figure 6 gives a summary of comparisons between hardware and software techniques. The table shows the security use, applicable techniques and remarks on major impact.

D. PROCEDURAL TECHNIQUES

Procedural techniques are required to set up, maintain and monitor the automated security system. They apply as well to protecting data in the form of hard copy reports as they do to protecting it in the form of backup tapes and disk packs, program listings, program decks, common data pools, and user ID-lists and passwords. They are needed to establish the manual as well as the automated methods by which the four functions of security are accomplished. These four functions are classifying and declassifying data, providing the means to safeguard the data, providing for proper accountability, and allowing the dissemination of the data on the basis of a need-to-know. [Wasserman 69]

The security that is obtained in any system utlimately rests on the responsibility and trustworthiness of the individuals who are associated with it. There are, therefore, two primary procedural techniques that transcend

Hardware-Software Techniques Comparison

Security Use	Automated Technique		Remarks
	Hardware	Software	
Limit input/output to executive software	Processor-mode, privileged instruction set	Macro calls, edit programs to eliminate I/o instructions	Hardware technique provides cost-effective on-line check
Limit programs and data to an execute controlled memory block	Memory bounds control	Boundary maps, edit programs to eliminate direct address use	Hardware technique provides cost-effective on-line check
Limit access to programs and data to predetermined read/write execute.	Read/write/execute bit logic. DAM read only lock	User rights profile table, edit programs to eliminate unauthorized calls	Hardware technique provides cost-effective on-line check
Provide for processing and proper routing of I/O requests.	I/O control and mask registers, interrupt logic	Software polling routine	Hardware technique provides cost-effective on-line capability
Provide failsafe mode for hardware failures.	Parity check circuits, redundant key circuits, code redundancy	On-line software diagnostics	Hardware technique provides cost-effective on-line capability
Check security level of programs and data against access or processing request security level	Security control flag bits	User rights profile table check	Software technique provides cost-effective on-line capability

Figure 6

Provide for insulation of user programs and data in main memory	Key word memory register, dedicated memory, memory partitioning ports, associative memory	Macro calls, edit programs to elinimate direct address use	Hardware techniques provide cost-effective on-line check
Provide main memory residue data erase	Main memory block erase, IOCP block erase	Executive erases data before returning memory block for allocation	Hardware technique only necessary in CPU limited systems
Provide for proper routing of data in IOCP	Unit address register I/O channel/No. character check, channel number check logic, I/O answer back check.	Software answer back subroutine	Less processing overhead in hardware techniques, less procurement cost in software technique
Provide check of security level of data against security level of channel	I/O security level register	Channel profile table check	Software more cost-effective
Secure communications channels	Special purpose crypto device	Software crypto program	
Verify identification of terminal users	Hang up and dial, key pattern generator	Code word entry and check	Software less cost but not as reliable as hardware
Provide for detection of unauthorized change to data or programs	DAM check sum logic	Check sum software routine	More overhead for software routine
Provide for failsafe mode for software failure	Processor mode privileged instruction set memory bounds	On-line diagnostics security monitor program	Both techniques are needed to provide effective control

Figure 6 cont.

Isolate support users from programs and data	Processor mode, memory bounds	Relocatable bootstrap, redundant coding, module dialogue, program interpretation	Software techniques have high overhead
Limit access to files or data elements	Security control flag bits	File classification code, file access rights, file access profile, data element classification code, data element profile	Software needed to provide effective control
Maintain security log		Document log, security program, file or data element log	No hardware technique available

Figure 6 cont.

the four functional area and apply equally to every aspect of system activity. The first is to guarantee that the requirement to set up, maintain and monitor the system is accomplished only by those designated to perform the indicated function. The second is the formal establishment by law of personnel responsibility for the safeguarding and dissemination of classified data. Each person is responsible to safeguard classified data or programs made available tc him for the performance of his official duties and to limit dissemination of that data to only those with proper security clearance and need-to-know. These two principle procedures are implemented formally in all government and in several commercial systems reviewed.

1. Classifying and Declassifying Procedures

Procedures are available that allow the assignment of security classification to designated individuals at the file and program level. In some systems the user profile table specifically allows the authority to classify/declassify given files. In all systems, the ability to classify/declassify files and programs is only permitted to the individual of highest level security authority in the installation, usually the system security officer.

2. Safeguarding Procedures

All systems use secure area protection for central computer areas and remote classified terminals. Access lists are maintained to allow entry into these areas. The establishment of such access lists is the designated responsibility of the system security officer through formal submittal to the security authority. In addition, this officer is charged with the establishment and maintenance of user/terminal profile tables that provide the authority to access data and files and the assignment of personal ID

and/or code words to authorized users. These should preferably be assigned in a random manner and at random intervals.

3. Accountability

Procedures are available to provide periodic review of access logs, security monitor logs, record counts/check totals and file logs. Such reviews are the responsibility of the system security officer or data adminstrator and are called for at stated periodic times. Document signout procedures similar to that used in manual systems for classified data are used for hard copy classified material. An inventory of all hard-copy classified material back-up tapes disk, packs, program listings and card-decks is conducted on a periodic basis.

4. Dissemination

The dissemination of data in automated systems is based on user/terminal profile tables in some systems and on the use of access lists in others. The system security officer is responsible for the preparation and maintenance of the table and lists. In systems that use passwords or code words, procedures are established for the dissemination of the current codes on a periodic basis to those authcrized users. These procedural techniques are applicable to any system, since they are a common requirement for providing adequate protection. [Wasserman 69]

E. MANUAL AND AUTOMATED PROCEDURAL COMPARISONS

A comparison of procedures used in automated systems versus those that are used in manual systems provides a method to judge the relative value of automated techniques that are common to both as well as those which are analogous. In both automated and manual military systems

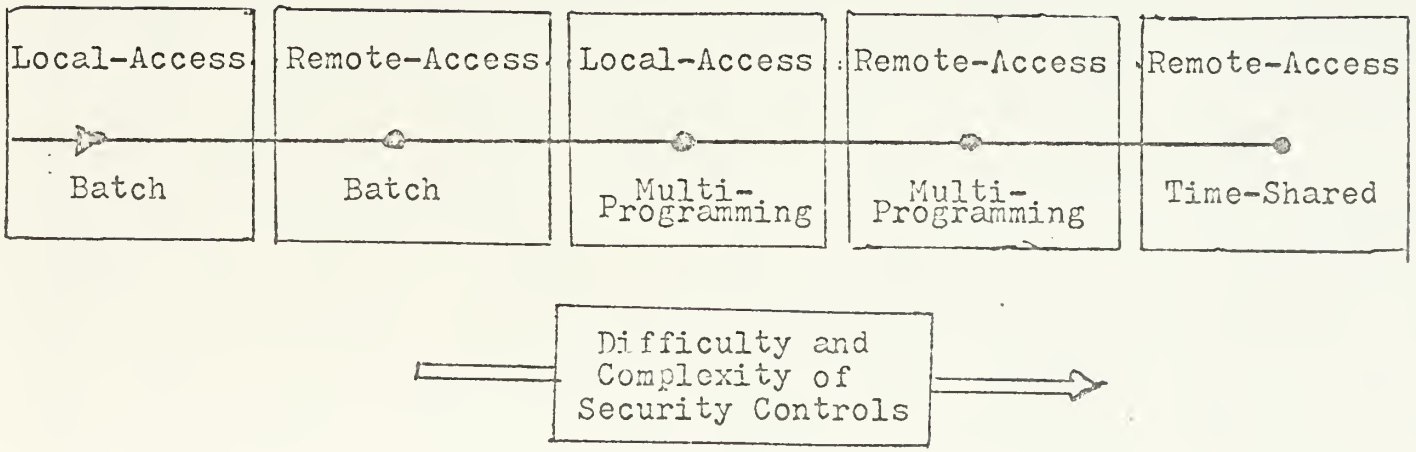

Local-Access
Remote-Access
Local-Access
Remote-Access
Remote-Access
Batch
Batch
Multi-Programming
Multi-Programming
Time-Shared
Difficulty and Complexity of Security Controls

Figure 7

procedural techniques are used to (1) secure access where classified data is used, (2) to assure proper clearance of personnel, (3) to classify, access, disseminate, and control classified data, and (4) to protect classified data during transmission by cryptographic secure communication lines. The addition of automated techniques to increase the reliability of these procedures could be viewed as an attempt to increase the security of automated systems over that of manual systems.

Analogous techniques used in the two systems are (1) data storage procedures, (2) data access procedures, (3) data access accounting, (4) storage check procedures and, (5) inventory procedures.

Data access procedures in a manual system are based upon access lists and personnel identification. In automated systems, access to data and files is based upon user/terminal profile tables and the requirement to submit the proper code word. Other techniques have been suggested such as fingerprint and voice code pattern generators.

Data access accountability in manual systems is performed by document sign-out. In automated systems, logs of file access by user or terminal identification can be kept automatically. Daily safe checks are used in manual systems to insure storage integrity. In automated systems, the access logs and security program reports can be reviewed as often as desired. Periodic inventory is used in manual systems to insure documents have not been lost or stolen. In automated systems the files are reviewed periodically, check sum totals are used to insure data integrity, and all security logs are reviewed. The conclusion is that the automated system with the use of modest security techniques, can provide a greater level of security than is possible in a manual system.

VI. CONCLUSIONS

It is important to understand what present technology can and cannot do in protecting classified information in a resource sharing system. Present technology offers no way to absolutely protect information or the computer operating system itself from all security threats posed by the human beings around it. As a consequence, procedural and administrative safeguards must be applied in resource-sharing computer centers to supplement the protection available in the hardware and software.

Security control in a computer system, especially a resource sharing one, is a system design problem, and solutions to it must be based on a systems point of view. The future of data bank security lies in designing a system with adequate protection which is not so complex or expensive as to discourage its use. In principle, the number, type, and depth of security controls in a system should depend on the sensitivity of the information in the system, on the class of users being served, on the geographical distribution of the system, on the nature of the service that the system provides its users, and on the operational situation that the system supports.

The system designer must be aware of the totality of potential leakage points in any system in order to create or prescribe techniques and procedures to block entry and exploitation. The security problem of specific computer systems must be solved on a case-by-case basis employing the best judgement of a team consisting of system programmers, technical, hardware, and communications specialists, and security experts.

BIBLIOGRAPHY

ALLEN 68 B. Danger Ahead... Safeguard Your Computer (HARVARD BUSINESS REVIEW) Nov-Dec. 1968

ANONYMOUS 69 The Technology of Computer destruction: A do it yourself handbook for would-be saboteurs. (SEED MAGAZINE) Underground, 1969

ASTOR 71 S.D. An Investigator Talks of Embezzlement and Robbery: (THE OFFICE) september 1971

AVCO COMPUTER SERVICES 71 The most complete bibliography to date.Contains a number of computerworld -horror story- articles. (AVCO) 1971

BABCOCK 67 J.D. A Brief Description of Privacy Measures in the RUSH Time-Sharing System: A bief summary of the ABC time sharing system. (SPRING JOINT COMPUTER CONFERENCE) , 1967

BARAN 65 P. Communications, computers and people: A well thought out general discussion of the privacy problem which overlaps somewhat with Barans testimony before the Gallagher subcommittee. Some specific proposals are presented to deal with the problem. (AFIPS, FALL JOINT COMPUTER CONFERENCE) 1965

BATES 70 W.S. Security of Computer Based Information Systems: (DATAMATION) Vol.16 No.5 p60-65 May 1970

BEARDSLEY 72 C.W. Is Your Computer Insecure?: Hardware and software techniqes to prevent computer disasters. (IEEE SPECTRUM), January 1972

BINGHAM 65 H.W. Security Techniques For EDP of Multi-Level Classified Information: Document, RADC-TR-65-415, 1965

BRANSTAN 73 D.K. Privacy and Protection in Operating Systems: A workshop report on designing a secure operating system on present hardware, new hardware protection facilities, weaknesses of current systems protection features and methods of continually monitoring a secure system. (COMPUTER MAGAZINE), January 1973

BUSINESS WEEK 70 Security Men Thrive on the wages of Fear. (BUSINESS WEEK) NO.2129 p112-114 June 20, 1970

CARROL 71 J.M. Multi-Dimensional Security Program For A Generalized Information Retrieval System: GIRS is a generalized information retrieval system which permits the creation and modification of a data base, as well as the retrieval of specified data from the base. (FALL JOINT COMPUTER CONFERENCE) 1971

CHU 71 A.L.C. Computer Security: The Corporate Achilles Heel: This article is addressed to the computer center manager and his management staff. It warns of the many

hazards of computer center operation and suggests means for obviating problems. (BUSINESS AUTOMATION) No.18 Feb. 1971

COMPUTER WORLD 70 Limiting Access to Centers Called a Major Problem (COMPUTER WORLD) Vol-4 NO.25 June 24, 1970

CONTROL DATA 66 Control Data 6400/6600. Computer systems reference manual, No.60100000, 1966

CONWAY 72 R.W. On The Implementation Of Security Measures In Information Systems: The security of an information system may be represented by a model matrix whose elements are decision rules and whoserow and colum indices are users and data items respectively. (COMM OF THE ACM), April 1972 Vol 15 Number 4

CONWAY 72 R. Selective Security Capabilities In ASAP: A file Management System. (SPRING JOINT COMPUTER CONFERENCE) 1972

CORBATO 65 F.J. Introduction and Overview of the Multics System: AFIPS Conference Procedings. (FALL JOINT COMPUTER CONFERENCE) 1965

CROSS 67 R.F. Safeguarding Classified Information: (INDUSTRIAL SECURITY) 1967

DALEY 65 R.C. A General Purpose File System for Secondary Storage: AFIPS Conference Proceedings. (FALL JOINT COMPUTER CONFERENCE) 1965 p.213-229

DATA PROCESSING DIGEST 70 Computer Security: (DATA PROCESSING DIGEST) 1970

DATA SYSTEMS NEWS 70 Protecting Your Computer's Security: (DATA SYSTEMS NEWS) 1970

DAVIS 71 A.G. Security of the Computer Center: (INDUSTRIAL SECURITY) 1971

DAVIS 71 G. Auditing and EDP, American Institute of Certified Public Accountants: A very important book for auditors, who are and will be forced to become involved with the internal workings of computers.

DAVIDSON 69 T.A. Computer Information Privacy : (THE OFFICE) 1969

DEAN 71 A. jr. Data Privacy And Integrity Requirements for On-line Data Management Systems ACM - SIGFDET Workshop on Data Description, Access and Control

DEFENSE INTELLIGENCE AGENCY 69 Security of Classified Information in the Dis Ansrs System: Official Use Only. (DIA: C-3663 MS5) Feb. 1969

DENNING 71 P.J Third Generation Computer Systems: Denning has long been a proponent of the fact that there is a viable and consistent theory of protection. He mentions many universal concepts in this paper. (COMPUTING SURVEYS) Vol.3 No.4 Dec. 1971

DENNIS 65 J.P. Segmentation And The Design Of Multi-programmed Computer Systems: (ACM JOURNAL) Vol. 12 No. 4 Oct. 1965 p. 598

DENNIS 66 J.P. Programming Semantics For Multiprogrammed Computation: A pioneering work, introducing the concepts of capabilities and c-list upon which most present day theories of protection are based. (COMMUNICATION OF THE ACM) Mar. 1966

DENNIS 66 R.L. Security In The Computer Environment: Results of a 1965 Conference of Research Security Administrators. SDC-SP 2440-00-01, AD640648 Aug. 1966

DEPARTMENT OF DEFENSE 70 Industrial Security Manual For Safeguarding Classified Information: DOD 5220 22M, US Government Printing Office April 1970

DEVITT 70 R.G. Cut Expenses By Taking Care Of Your Tape: (COMPUTER DECISION) Vol.2 No.10 1970

DIJKSTRA 65 E.W. Cooperating Sequential Processes: (DEPARTMENT OF MATHEMATICS, TECHNOLOGICAL UNIVERSITY, EINDOVEN, THE NETHERLAND) 1965

DIJKSTRA 68 E.W. The Structure of The Multiprogramming System: Dijkstra has accomplisned valuable work in the theory and solutions of integrity problems. He is responsible for the term "deadly embrace" by which deadlocks can cause paralysis of the system. (COMMUNICATION OF THE ACM) Vol. 22No. 5 May 1968 pp. 341-346.

DILLON 66 G.M. How Much Protection For Magnetically Recorded Data?: (SYSTEMS & PROCEDURES JOURNAL) Sept.-Oct. 1966

DONATI 71 F.R. Computers and Catastrophes: (DATA MANAGEMENT) Dec. 71

DONNELLY 69 G.J. Non-Interruptible Electric Power For A Large Computer System: (ELECTRICAL CONSTRUCTION DESIGN) 1969

ELECTRICAL WORLD 70 Computer Power In Small Packages: (ELECTRICAL WORLD) Vol.173 No.1 Jan. 1970

ELECTRONICS REVIEW 69 Providing The Right Environment (ELECTRONICS REVIEW) Vol.185 No.22 Nov. 1969

ELLIS 68 W. B. Security Procedures for the Rye System: Official Use Only. (NSA: C924) Dec. 1968

ENGER 67 I. Automatic Security Classification Study: Report of an investigation of the feasibility of using computers to assign the proper government security classification to textual material. The "correctness" was only 54%, but the technique used did show promise for futher research. RADC TR 67-472 Oct. 1967

EVANS 67 D.C. Address Mapping and the Control of Access In An Interactive Computer: (AFIPS CONFERENCE PROCEEDINGS,SPRING JOINT COMPUTER CONFERENCE) 1967

FACTORY INSURANCE ASSOCIATION 70 Recommended Good Practice For The Protection Of Electronic Data Processing and Industrial Automation: A must for installation of physical security and fire preventive efforts. (FACTORY INSURANCE ASSOCIATION) 1970

FANWICK 67 C. Computer Safeguards: How Safe Are They? (SDC MAGAZINE) Vol.10 No.7&8 Jul-Aug. 1967

FANWICK 66 C. Maintaining Privacy Of Computerized Data: (SP2647, SYSTEM DEVELOPMENT CORPORATION) Dec. 1966

FEIGE 69 E.L. Protection of Privacy Through Microaggregation: (SOCIAL SYSTEM RESEARCH INST.) Univ. of Wisc. 1969?

FEISTAL 70 J. Cryptographic Coding For Data Bank Privacy: (IBM CORPORATION RC282M) March 1970

FELSER 71 G.M. How Much Longer Will Your Humpty Dumpty Stay On The Wall?: An article for risk managers and those about to think of EDP insurance. (EDP DISASTER PROTECTION WORKSHOP, A 28TH INTERNATIONAL CONFERENCE) 1971

FENWICK 71 W.A. Marketing Edp Service: Reviewing The Legal Considerations: A misnomer. The author is really talking about security measures to protect the confidentiality of data. (COMPUTERS AND AUTOMATION) Nov. 1971

FOSTER 71 C.C. Data Banks; A Position Paper: A summary of what can go wrong (machine failure, logical errors, eavesdropping, wiretapping) and what to do about it (individual rights of privacy protection and government regulation). (COMPUTERS AND AUTOMATION) Mar. 1971

FOX 69 C. Security Controls in the ADEPT-50 Timesharing System: Security control based on a set theoretical model of access rights. (FALL JOINT COMPUTER CONFERENCE) 1969

FREED 69 R.N. Computer Fraud; A Management Trap: (BUSINESS HORIZONS) June 1969

FRIEDMAN 70 T.D. The Authorization Problem In Shared Files: The problem of sharing information, yet

providing proper authorization is reviewed. A model of a secured file system provides the basis for much current work in the field. Friedman rejects the hierarchical approach to authorization in favor of compartments categorization of access rights. (IBM SYSTEMS JOURNAL) No.4 1970 p.258-280

FUSS 68 E.L. Potential Solutions In Computer Security Control:(SEMINAR-AMERICAN SOCIETY INDUSTRIAL SECURITY) Sept. 1968

GAINES 72 R.S. An operating System Based On The Concept Of A Supervisory Computer: (COMMUNICATIONS OF THE ACM) Vol.15 No.3 Mar. 1972

GARRISON 70 W.A. Privacy And Security In Data Banks: Discusses some possible solutions both legal and technical to insure privacy and security of imformation. (UNIVERSITY OF TEXAS PRESS) 1970

GELLMAN 71 H.S. How The Computer Can Be Used To Rob You Blind: (RISK MANAGEMENT) Aug.Sept. 1971

GELLMAN 71 H.S. Using The Computer To Steal: (COMPUTERS & AUTOMATION) April 1971 p.16

GIRDANSKY 71 M.B. Data Privacy: Cryptology and The Computer At IBM Research: An interesting study on what researchers are doing to devise "unbreakable" codes, and how many classical approaches to encipherment are easily compromised. The paper discusses "Lucifer", a hardware encryption device. (IBM RESEARCH REPORTS) Vol. 7 No. 4 1971

GLASSER 67 E.L. A Brief Description of Privacy Measures in the Multics Operating System: The problems of maintaining information privacy in a multi user, remote access system. discussion of the complexities. (SPRING JOINT COMPUTER CONFERENCE) 1967

GLASSER 68 E.L. The Safeguarding Of Information: A User's View (PROCEEDING IFIP CONGRESS) 1968 Supplement Booklet 1, pp.3-6.

GLASSER 65 E.L. System Design Of A Computer For Time Sharing Applications: (AFIPS CONFERENCE PROCEEDINGS, FALL JOINT COMPUTER CONFERENCE) 1965

GODBOUT 71 W. Computer Theft By Computer: (SECURITY WORLD) May 1971

GRAHAM 68 R.M. Protection in an Information Processing utility: One of the critical problems in the design of an information processing utility that permits flexible sharing of user information is privacy. One solution for this problem is discussed. (COMM OF THE ACM) Vol 11 Number 5 May 1968

GRAHAM 72 G.S. Protection, Principles And Practice: Graham and Denning have analyzed most existing systems

and find they fit the protection model that is described. The work draws heavily on the theories of the "princeton" school. (AFIPS CONFERENCE PROCEEDINGS,SPRING JOINT COMPUTER CONFERENCE) May 1972

GRAHAM 68 R.M. Protection In An Information Prccessing Utility: Describe the "rings of protection" in MULTICS. (CACM) Vol.11 No.5 May 1968

GRANT 69 C.B.S. Will Students Wreck Your Computer Center?: (DATA PROCESSING) May 1969

GUIDE-SHARE 70 Data base Management System Requirements: An important document that outlines idealized requirements for data base management. Security and integrity play a dominant role. (JOINT GUIDE-SHARE DATA BASE REQUIREMENT GROUP) Nov. 1970

HABERMANN 69 A.N. Prevention of system Deadlocks (COMMUNICATIONS OF THE ACM) Vol.12 No.7 1969

HANSEN 71 M.H. Insuring Confidentiality of individual Records in Data Storage and Retrieval For Statistical Purpose: (AFIPS CONFERENCE PROCEEDINGS,FALL JOINT COMPUTER CONFERENCE) Nov. 1971

HANSEN 70 M.H. Some Aspects Of Confidentiality In Information Systems: (PAPERS FROM THE EIGHTH ANNUAL CONFERENCE OF THE URBAN REGIONAL INFORMATION SYSTEM ASSOCIATION) Sept. 1970

HANSEN 70 P.B. The Neucleus Of A Multiprogramming System: (COMMUNICATIONS OF THE ACM) Vol. 13 No. 4 April 1970

HIRCHFIELD 71 R.A. Security In An On-Line System; A Primer For Management: Management beware, security is comprehensive control. Most installations dont have it. (COMPUTERS AND AUTOMATION) Sept. 1971

HOFFMAN 69 L.J. Computers and privacy; A Survey: The "classic" survey on this subject. (COMPUTING SURVEY) Vol.1 No.2 June 1969

HOFFMAN 70 L.J. Getting A Personal Dossier From A Stastical Data Bank: An interesting example of how tc impute information by indirect means from innocent files, (DATAMATION) May 1970 p.74-75

HOFFMAN 70 L.J. The Formulary Model For Access Control And Privacy: (STANFORD LINEAR ACCELERATOR CENTER (SLAC) REPORT) No.17 May 1970

HOFFMAN 71 L.J. The Formulary Model For Flexible Privacy And Access Control: These two papers present a model for interfacing user access control and data by means of ccded procedures called formularies. (AFIPS CONFERENCE PROCEEDINGS, FALL JOINT COMPUTER CONFERENCE) Nov. 1971

HOWES 71 P.R. EDP Security Is Your Guard Up? A "worry" paper for managers. (AMA MANAGEMENT REVIEW) July 1971

HSIAO 68 D.K. A File System For A Problem Solving facility: Hsiao discusses the use of an "authority item" which allows protection below the file level. The system is baesd on a multi-list file structure. (PHD DISSERTATION IN ELECTRICAL ENGINEERING, UNIV. OF PENNA.) 1968

IBM 68 The Considerations Of Data Security In A Computer Environment: A widely distributed monograph on data security; it helped bring attention to some of the needs and the problems. (IBM-DATA PROCESSING DIVISION) 1968

IBM 69 Data Processing Techniques For Management Control Of Electronic Data Processing: IBM Manual F20-0006-0 Sept. 1969

IBM 71 Guidelines For Protection & Control In A Computer Environment: This is one of the very best manuals on the subject of how to develop and implement a comprehensive computer security program. See your local IBM represenative Its better than any thing else on the market and costs less. (IBM CORPORATION, POUGHKEEPSIE, NEW YORK, D P HEADQUARTERS, ibm FIELD SYSTEMS CENTER) 1971

INDUSTRIAL SECURITY 69 Problems and Potential Solutions In Computer Control: (INDUSTRIAL SECURITY) April 1969

INDUSTRY WEEK 70 Some Tips On Computer Security: (INDUSTRY WEEK) Aug. 1970

INDUSTRY WEEK 70 Computer Security Is Sensitive Area: (INDUSTRY WEEK) Vol.167 Oct. 1970

JACOBSEN 70 R.V. Providing Data Security: (DATAMATION) Vol.17 No.6 June 1970

JACOBSEN 71 R. V. Cost Effectiveness Of Security Measures: Jacobsen is a highly competent consultant on computer security. He has developed a sound approach to security design and is especially knowledgable about physical security. (ABA NATIONAL AUTOMATION CONFERENCE) May 71

JACOBSON 70 R.V. Planning For Backup Facilities: (COMPUTER SERVICES) Vol-2 n0.3 p22-29 May-June 1970

JACOBSON 70 R.V. Providing Data Security: (AUTOMATION) June 1970

KAHN 67 D. The Code Breakers: The classic book for those interested in cryptology. (THE MACMILLAN COMPANY) 1967

KJELDAAS 69 M. Security In Software: (DATA PROCESSING DIGEST) Sept. 69

KRAUSS 69 L. I. Administering and Controlling The Company Data Processing Function: (ENGLEWOOD CLIFFS, NEW JERSEY, PRENTICE HALL) 1969

KRAUSS 72 L.I. SAFE: Security Audit and Field Evaluation For Computer Facilities and Information Systems: A computer security checklist. (FIREBRAND, KRAUSS and COMPANY, PUBLISHERS) Mar. 1972

LAMPSON 69 B. W. Dynamic Protection Structures: (AFIPS CONFERENCE PROCEEDINGS, FALL JOINT COMPUTER CONFERENCE) Vol.35 pp.27-38

LAUREN 70 R. H. Reliability of Data Bank Records: (DATAMATION) May 1970 pp.88-89.

LEARSON 72 T. V. Keynote Address: 1972 Spring joint Computer Conference: IBM is committed to spending $40 million over four years for data security. RSS has been implemented in four test sites, and will be available in version 21 of OS/MVT. All results will be in the public domain. (SPRING JOINT COMPUTER CONFERENCE PRESS RELEASE) May 1972

LESSER 68 V. R. A Multi-level Computer Organization Designed To Separate Data-accessing From Computation: (TECH. REP. CS90-COMPUTER SCIENCE DEPARTMENT, STANFORD UNIVERSITY) Mar. 1969

LINDE 69 R. C. The Adept-50 Time-sharing System: Important for understanding a significant implementation of protection. (AFIPS CONFERENCE PROCEEDINGS,FALL JOINT COMPUTER CONFERENCE) Vol.35 1969 pp35-50.

LINDE 69 R. Security Controls in the ADEPT-50 Timesharing System: Security control based on a set theoretical model of access rights. (FALL JOINT COMPUTER CONFERENCE) 1969

LUTTER 70 F. H. Protecting The Data Base: (ADMINISTRATIVE MANAGEMENT) Vol.31 Nov. 1970

MARTIN 67 J. Design Of Real Time Computer Systems: (ACCURACY AND SECURITY) Chap.35 1967

MAXWELL 72 W.L. On The Implementation Of Security Measures In Information Systems: The security of an information system may be represented by a model matrix whose elements are decision rules and whose row and colum indices are users and data items respectively. (COMM OF THE ACM), April 1972 Vol 15 Number 4

MCCAHILL 71 F. X. Jr. Avoid Losses Through Risk Management: Talks about protection from the point of view of the corporate insurance manager. The concepts can be applied to computer security management. (HARVARD BUSINESS REVIEW) May-June 1971

MILLER 71 A. R. The Assault on Privacy: This book is one of the definitive volumes on computers and privacy. It is

listed here because of its importance, even though this bibliography excludes citations exclusively devoted to privacy. (UNIVERSITY OF MICHIGAN PRESS) 1971

MINTZ 70 H. K. Safeguarding Computer Information: (SOFTWARE AGE) May 1970.

MODERN DATA 73 Security In The Palm Of Your Hand: Identimation's unique I.D. system is a hands down winner in the security Department (MODERN DATA) Feb. 1973

MOLHO 70 L. M. Hardware Aspects of Secure Computing: This paper is about hardware aspects of controlled access time-shared computing. (AFIPS CONFERENCE PROCEEDINGS,SPRING JOINT COMPUTER CONFERENCE) SDC sp3453 Dec. 1969.

MORGAN 72 H.L. On The Implementation Of Security Measures In Information Systems: The security of an information system may be represented by a model matrix whose elements are decision rules and whose row and colum indices are users and data items respectively. (COMM OF THE ACM), April 1972 Vol 15 Number 4

NOTZ 71 W. A. An Experimental Application Of Cryptography To A Remotely Accessed Data System: Describes "Lucifer" a hardware encryption and decoding device attached to a time shared 360/67.

O'CONNELL 71 M. L. A File organization Using Multiple Keys:(AFIPS CONFERENCE PROCEEDINGS,SPRING JOINT COMPUTER CONFERENCE) 1971

O'CONNOR 71 R. A. The Software Techniques Or Data Security: (PAPER TO THE MIDLANDS CHAPTER,ACM) Sept. 1971

PETERS 67 B. Security considerations in a Multiprogrammed Computer System: A discussion of the degree of capability and limitation, of security measures, commensurate with the material or operation being secured. (SPRING JOINT COMPUTER CONFERENCE) 1967

PETERSON 67 H. E. System Implications of Information Privacy: The 1967 SJCC highlighted computer security as a major subject. (AFIPS CONFERENCE PROCEEDINGS,SPRING JOINT COMPUTER CONFERENCE) Vol 30 1967 pp.291-300.

PFOFF 69 A. M. Structuring The Data Security Problem: (GUIDE 29 PROCEEDINGS) 1969

PLANNING RESEARCH CORPORATION 70 Pacer Multi-Level Security Program Design Specifications: Pacer is a military intelligence analysis system. It processes mixes of highly classified data, therefore security is a prime consideration in its design. It is implemented on Honeywell 6000 series equipment. (PRCWP0115) Nov. 1970

SCHROEDER 71 M.D. A Hardware Architecture For Implementing Protection Rings: Presented at the third annual ACM on

operating systems principles Oct. 1971

SCOMA 70 Jr. L. Security In The Computer Complex: (COMPUTERS AND AUTOMATION) Vol.19 No.1 Nov. 1970

SHANNON 49 C. E. Communication Theory Of Secrecy System: The theory of cryptology has not been significantly improved since this was published. (BELL TELEPHONE SYSTEM TECHNICAL JOURNAL) Vol.28 No.4 Oct. 1949

SKATRUD 69 R. O. A Consideration Of The Application Of Cryptographic Techniques To Data Processing: (AFIPS CONFERENCE PROCEEDINGS, FALL JOINT COMPUTER CONFERENCE) Vol.35 1969 pp.111-117

SMITH 71 J. L. The Design Of Lucifer, A Crytographic Device For Data Communications: (IBM CORPORATION) RC3326 Apr. 1971

SORENSON 72 J. L. Common Sense In Computer Security: There are many measures that can be taken to improve computer security. All are costly. But, taking a commcn sense approach to the problem may help keep the costs down. (JOURNAL OF SYSTEMS MANAGEMENT) Apr. 72

TASSEL 69 C. J. Information Security in A Computer Environment: (COMPUTERS & AUTOMATION) Jul. 1969

TAYLOR 70 R. L. Computer Data Protection: (INDUSTRIAL SECURITY) Vol.14 No.4 Aug. 1970

THORNE 71 J. F. Internal Control of Real-time Systems: (DATA MANAGEMENT) Jan. 71

TITUS 67 J.P. Security and Privacy: A half-dozen computer users and designers devoted two complete sessions of the spring joint computer conference in April tc their attempts to protect sensitive information in multiple access computers. (COMM OF THE ACM) Vol-10 Number 6 June 67

TURN 70 R. Security of Computerized Information Systems: (RAND P-4405) Jul. 1970

TURN 72 R. Privacy And Security In Data Bank Systems: Measures of effectiveness, costs, and protector-intruder interactions. A structural model of data bank systems is formulated, the roles of its elements in providing, or threatening, of data privacy and security are discussed. (RAND PUBLICATION) 1772

VAN TASSEL 69 D. Information Security In A Computer Environment: Information security can never become obsolete, but it is possible to take a few steps so that the cost to the penetrator will become high; hopefully higher than he is willing to pay. (COMPUTERS AND AUTOMATION), July 1969.

VAN TASSEL 69 D. Advanced Cryptographic Techniques For

Computers: (COMMUNICATIONS OF THE ACM) Dec. 1969

WARE 67 W. H. Security and Privacy in Computer Systems: Information leakage in a resource-sharing computer system. (AFIPS CONFERENCE PROCEEDING,SPRING JOINT COMPUTER CONFERENCE) Vol.30 1967 pp.287-290

WARE 67 W.H. Security and Privacy: Similarities and Differences.: Discussion of problems and implications of computer security and privacy. (SPRING JOINT COMPUTER CONFERENCE) 1967

WARE 70 W. H. Computer Data Banks And Security Control: (RAND P-4329) 1970

-- Reference Deleted --

WASSERMAN 69 J. J Plugging The Leaks In Computer Security: (HARVARD BUSINESS REVIEW) Vol.7 Sept. 1969 pp119-129

WASSERMAN 70 J J. Protecting Your Computer's Security: (DATA SYSTEM NEWS) Feb.1970

WEISSMAN 67 H. Programming Protection: What Do You Want To Pay?: (SDC MAGAZINE) Vol.10 No.8 Aug. 1967

WEISSMAN 69 C. Security Controls in the ADEPT-50 Timesharing System: Security control based on a set theoretical model of access rights. (FALL JOINT COMPUTER CONFERENCE) 1969

WELKE 71 L. A. What About Security?- On Centers: (MODERN DATA) Sept. 1971

WESSLER 71 J. Physical Security-Facts and Fancies: (DATAMATION) Jul.1971

WESTIN 67 A. F. Privacy and Freedom: (ATHENCUM PRESS,NEW YORK) 1967

YOURDON 72 E. Reliability of Real-time Systems: A six part series of articles (from a book of the same title) explores why and how systems fail. (MODERN DATA) Jan.-Jun.1972

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 212 Naval Postgraduate School Monterey, California 93940	2
3.	Professor G. L. Barksdale, Jr., Code 72 Computer Science Group Naval Postgraduate School Monterey, California 93940	1
4.	LCDR R. M. Hanna, Code 53Hq Naval Postgraduate School Monterey, California 93940	1
5.	LT W. L. Lupton 2316 N. Beechwood Street Philadelphia, Pennsylvania 19132	1

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

ORIGINATING ACTIVITY (Corporate author)
Naval Postgraduate School
Monterey, California 93940

2a. REPORT SECURITY CLASSIFICATION
Unclassified

2b. GROUP

REPORT TITLE

A Study Of Computer Based Data Security Techniques

DESCRIPTIVE NOTES (Type of report and inclusive dates)
Master's Thesis; (June 1973)

AUTHOR(S) (First name, middle initial, last name)
William L. Lupton

REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
June 1973	75	142

a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
b. PROJECT NO.	
c.	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)
d.	

0. DISTRIBUTION STATEMENT

Approved for public release; distribution unlimited.

1. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
	Naval Postgraduate School Monterey, California 93940

3. ABSTRACT

This essay presents the results of a study which surveyed the various aspects of system security hardware, software, and procedural techniques in use in current and proposed automated systems. Its impetus is from the concern for security control that has been generated by the increasing number of time-sharing and resource-sharing systems. The intention is to present the designers, managers, programmers, system implementers, and operational personnel with a consolidated source of data concerning security techniques and with a tool to evaluate the data and select the techniques applicable to their respective security requirements.

DD FORM 1 NOV 65 1473 (PAGE 1)

S/N 0101-807-6811

Security Classification

A-31408

KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Data Security						
Computer Security						
Access Security						
Privacy/Security